



[illegible]

(2) 48  
(3) 78  
(4) 136

DECLARATIONS  
MTHSGFLOOR - greatest integer G<sub>floating</sub> routine  
MTHSGFLOOR\_R3 - greatest integer G<sub>floating</sub> routine

```
0000 1      .TITLE MTH$GFLOOR - Greatest integer routine for Gfloating
0000 2      .IDENT /1-001/ ; File: MTHGFLOOR.MAR
0000 3
0000 4
0000 5      *****
0000 6      *
0000 7      * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
0000 8      * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
0000 9      * ALL RIGHTS RESERVED.
0000 10     *
0000 11     * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
0000 12     * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
0000 13     * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
0000 14     * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
0000 15     * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
0000 16     * TRANSFERRED.
0000 17     *
0000 18     * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
0000 19     * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
0000 20     * CORPORATION.
0000 21     *
0000 22     * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
0000 23     * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
0000 24     *
0000 25     *
0000 26     *****
0000 27
0000 28
0000 29     ++
0000 30     FACILITY: Math Library
0000 31
0000 32     ABSTRACT:
0000 33
0000 34         This routine finds the largest integer less than the input
0000 35         value, i.e. it truncates toward negative infinity
0000 36         for data type Gfloating.
0000 37
0000 38     ENVIRONMENT: User Mode, AST Reentrant
0000 39
0000 40     --
0000 41     Author: John Sauter, Creation date: 27-JUL-1979
0000 42
0000 43     MODIFIED BY:
0000 44
0000 45     VERSION 00
0000 46     1-001 - Original, from MTH$DFLOOR.
```



```

0000 48      .SBTTL  DECLARATIONS
0000 49      ::
0000 50      INCLUDE FILES:
0000 51      ::
0000 52      ::
0000 53      ::
0000 54      EXTERNAL DECLARATIONS:
0000 55      ::
0000 56      .DSABL  GBL                      ; Prevent undeclared
0000 57                                          ; symbols from being
0000 58                                          ; automatically global.
0000 59      ::
0000 60      MACROS:
0000 61      ::
0000 62      ::
0000 63      ::
0000 64      EQUATED SYMBOLS:
0000 65      ::
0000 66      ::
0000 67      ::
0000 68      OWN STORAGE:
0000 69      ::
0000 70      ::
0000 71      ::
0000 72      PSECT DECLARATIONS:
0000 73      ::
00000000 74      .PSECT _MTH$CODE PIC, USR, CON, REL, LCL, SHR, -
0000 75      EXE, RD, NOWRT, LONG
0000 76

```

```
0000 78 .SBTTL MTH$GFLOOR - greatest integer G_floating routine
0000 79 :++
0000 80 : FUNCTIONAL DESCRIPTION:
0000 81 :
0000 82 : This routine finds the floor by truncating, and then if the
0000 83 : input value is negative and not an integer subtracting 1.
0000 84 :
0000 85 : CALLING SEQUENCE:
0000 86 :
0000 87 : CALL result_int.wg.v = MTH$GFLOOR (input.rg.r)
0000 88 :
0000 89 : INPUT PARAMETERS:
0000 90 :
00000004 0000 91 : input_addr = 4 ; address of the G_floating number
0000 92 : ; to get the floor of
0000 93 :
0000 94 : IMPLICIT INPUTS:
0000 95 :
0000 96 : NONE
0000 97 :
0000 98 : OUTPUT PARAMETERS:
0000 99 :
0000 100 : NONE
0000 101 :
0000 102 : IMPLICIT OUTPUTS:
0000 103 :
0000 104 : NONE
0000 105 :
0000 106 : FUNCTION VALUE:
0000 107 : COMPLETION CODES:
0000 108 :
0000 109 : the G_floating value of the greatest integer
0000 110 :
0000 111 : SIDE EFFECTS:
0000 112 :
0000 113 : NONE
0000 114 :
0000 115 :--
0000 116 :
0000 117 :.ENTRY MTH$GFLOOR, ^M<R2, R3> ; entry point
0002 118 :
52 52 08 50 04 BC 50FD 0002 119 : MOVG @input_addr(AP), R0 ; R0/R1 = input argument
52 52 08 00 50 54FD 0007 120 : EMOVG R0, #0, #1, R2, R2 ; R2/R3 = fraction_part (arg)
52 52 08 50 52 42FD 000E 121 : SUBG2 R2, R0 ; R0/R1 = integer_part (arg)
0012 122 :
0012 123 : BGTR 40$ ; if > 0, have correct answer
0014 124 :
52 53FD 0014 125 : TSTG R2 ; look at fraction part
04 18 0017 126 : BGEQ 40$ ; if > 0 then 0 < input < 1 and
0019 127 : ; we have the correct answer
0019 128 : ; if = 0 then input was integer
0019 129 : ; and we have correct answer
50 08 42FD 0019 130 :
001D 131 : SUBG2 #1,R0 ; subtract 1 from truncated
001D 132 : ; negative non-integer
001D 133 :
04 001D 134 40$: RET
```

```
001E 136 .SBTTL MTH$GFLOOR_R3 - greatest integer G_floating routine
001E 137 :++
001E 138 : FUNCTIONAL DESCRIPTION:
001E 139 :
001E 140 : This is the JSB entry point to MTH$GFLOOR.
001E 141 :
001E 142 : CALLING SEQUENCE:
001E 143 :
001E 144 : JSB result_int.wg.v = MTH$GFLOOR_R3 (input.rg.v)
001E 145 :
001E 146 : INPUT PARAMETERS:
001E 147 :
001E 148 : R0 and R1 contain the input value
001E 149 :
001E 150 : IMPLICIT INPUTS:
001E 151 :
001E 152 : NONE
001E 153 :
001E 154 : OUTPUT PARAMETERS:
001E 155 :
001E 156 : NONE
001E 157 :
001E 158 : IMPLICIT OUTPUTS:
001E 159 :
001E 160 : NONE
001E 161 :
001E 162 : FUNCTION VALUE:
001E 163 : COMPLETION CODES:
001E 164 :
001E 165 : the G_floating value of the greatest integer
001E 166 :
001E 167 : SIDE EFFECTS:
001E 168 :
001E 169 : NONE
001E 170 :
001E 171 :--
001E 172 :
001E 173 MTH$GFLOOR_R3:: : entry point
001E 174 :
52 52 08 00 50 54FD 001E 175 EMOVG R0, #0, #1, R2, R2 : R2/R3 = fraction_part (arg)
50 52 42FD 0025 176 SUBG2 R2, R0 : R0/R1 = integer_part (arg)
0029 177 :
0029 178 BGTR 40$ : if > 0, have correct answer
002B 179 :
52 53FD 002B 180 TSTG R2 : look at fraction part
04 18 002E 181 BGEQ 40$ : if > 0 then 0 < input < 1 and
0030 182 : we have the correct answer
0030 183 : if = 0 then input was integer
0030 184 : and we have correct answer
50 08 42FD 0030 185 :
0034 186 SUBG2 #1,R0 : subtract 1 from truncated
0034 187 : negative non-integer
0034 188 :
0034 189 40$: RSB
0035 190
0035 191 .END
```



MTH\$GFLOOR  
Symbol table

N 12  
- Greatest integer routine for G\_floatin 16-SEP-1984 01:27:29 VAX/VMS Macro V04-00 Page 5  
6-SEP-1984 11:23:41 [MTHRTL.SRC]MTHGFLOOR.MAR;1 (4)

INPUT\_ADDR = 00000004  
MTH\$GFLOOR 00000000 RG 01  
MTH\$GFLOOR\_R3 0000001E RG 01

+-----+  
! Psect synopsis !  
+-----+

PSECT name	Allocation	PSECT No.	Attributes												
ABS	00000000 ( 0.)	00 ( 0.)	NOPIC	USR	CON	ABS	LCL	NOSHR	NOEXE	NORD	NOWRT	NOVEC	BYTE		
MTH\$CODE	00000035 ( 53.)	01 ( 1.)	PIC	USR	CON	REL	LCL	SHR	EXE	RD	NOWRT	NOVEC	LONG		

+-----+  
! Performance indicators !  
+-----+

Phase	Page faults	CPU Time	Elapsed Time
Initialization	29	00:00:00.11	00:00:01.10
Command processing	117	00:00:00.44	00:00:03.29
Pass 1	76	00:00:00.50	00:00:02.14
Symbol table sort	0	00:00:00.00	00:00:00.00
Pass 2	45	00:00:00.40	00:00:02.01
Symbol table output	2	00:00:00.01	00:00:00.01
Psect synopsis output	2	00:00:00.02	00:00:00.02
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	273	00:00:01.48	00:00:08.62

The working set limit was 900 pages.  
2047 bytes (4 pages) of virtual memory were used to buffer the intermediate code.  
There were 10 pages of symbol table space allocated to hold 3 non-local and 2 local symbols.  
191 source lines were read in Pass 1, producing 11 object records in Pass 2.  
0 pages of virtual memory were used to define 0 macros.

+-----+  
! Macro library statistics !  
+-----+

Macro library name	Macros defined
_\$255\$DUA28:[SYSLIB]STARLET.MLB;2	0

0 GETS were required to define 0 macros.

There were no errors, warnings or information messages.

MACRO/ENABLE=SUPPRESSION/DISABLE=(GLOBAL,TRACEBACK)/LIS=LIS\$:MTHGFLOOR/OBJ=OBJ\$:MTHGFLOOR MSRC\$:MTHGFLOOR/UPDATE=(ENH\$:MTHGFLOOR)



0260

DIGITAL EQUIPMENT CORPORATION  
CONFIDENTIAL AND PROPRIETARY